APPROXIMATE MATCHING AS A KEY TECHNIQUE IN ORGANIZATION OF NATURAL AND ARTIFICIAL INTELLIGENCE

MARILYN MACK[1], GENNADI M. LAPIR[2], SIMON BERKOVICH[3]

[1] National Aeronautics and Space Administration,
Goddard Space Flight Center
Code 585 / Code 933
Greenbelt, MD 20771
marilyn.j.mack@gsfc.nasa.gov

[2] ZERES GmbH Universiteatstrasse 142,
44799 Bochum, Germany,
lapir@zeres.de

[3] Department of Computer Science,
School of Engineering and Applied Science,
George Washington University,
Washington, DC, USA,
berkov@seas.gwu.edu

The basic property of an intelligent system, natural or artificial, is "understanding". We consider the following formalization of the idea of "understanding" among information systems. When system I issues a request to system II, it expects a certain kind of desirable reaction. If such a reaction occurs, system I assumes that its request was "understood". In application to simple, "push-button" systems the situation is trivial because in a small system the required relationship between input requests and desired outputs could be specified exactly. As systems grow, the situation becomes more complex and matching between requests and actions becomes approximate.

The problem can be presented in vector quantization terms: given a large multidimensional space of all possible types of requests, like all possible sentences in a language, a certain subset of these possible requests - standard blocks - is selected and related to a purposeful set of reactions. When a request is issued to this system, it is matched in the quantized vector space to one of the standard blocks. The "understanding" of a request can be judged by the suitability of the produced action. Attempts at this scheme of simplification of the complex issue of understanding are quite common in Artificial Intelligence. Their basic objective is to minimize the complex implementation issues associated with knowledge interpretation and preparation. Ideally, this scheme should work in the following mode. A interrogating party issues a request; the information system responds with an interpretation of this request based on the best pattern selected from its vector quantization set. If the interrogating party is not completely satisfied with the answer, it may resubmit the request taking into account this answer. Hopefully, in a few iterations the desired understanding would be established. In fact, this is seemingly the way in which "understanding" is achieved in human conversation. So, in particular, this scheme may be used for a human giving oral orders to a computer system. However, this approach would impose rather strenuous requirements on the organization of computational processes. Namely, it requires a very large storage with content addressable access having approximate matching capabilities. Constructing a storage providing content addressable access with approximate matching capabilities represents one the challenges in computer science development. Efforts were directed primarily to software constructs resulting in file organizations with sophisticated searching facilities of exact matching. The problem of approximate matching seems more difficult.

The hardware construction of large content addressable storage with approximate matching facilities can be implemented using the principle of holography. From an engineering aspect, realization of this principle for storage of information poses certain technical difficulties, although there exists recent indication of progress and expectations. From a more general aspect, holography seems to be the type of organization best suited for the representation of human memory [1] [2]. Although, the physical mechanism that might be involved in human memory is uncertain, the organization, as we explore it here in the form of a computational model, gives interesting results irrespective of its mechanism. A possible realization based on a cellular automaton approach has been introduced in [3] and may be left for further consideration.

Current advances in VLSI technology, namely in the development of large capacity main memories, have enabled the successful realization of content-addressable storage. This allows fast searching facilities to be accommodated with bit attribute matrices. An effective method for approximate matching has been suggested in [4]. The system described is a search engine with content-addressable access to very large textual information systems based on N-grams with "vertical" representation of attribute bit-matrices. A

more detailed description of this system and a comparative evaluation of its searching facilities will be given elsewhere. Here we present one rather specific application of the capabilities of this search engine in the framework of the described concept of "understanding". This example is amazingly simple but has proved to be very effective. The application that we describe is related to what may be called "machine translation". In general, it is a very complicated subject that cannot be considered as completely successful from a practical viewpoint. However, in our case it is applied to a narrow area: machine translation of technical documentation. The suggested system contains a certain sufficiently large number of predetermined standard sentences. The "translation" process goes as follows: for each sentence composed by a technical writer, the developed system provides a number of permitted blocks: sentences representing what the search engine determines are the best or nearest approximations to the words and wording of the composed sentence. One of the standard sentences presented is chosen and the process continues. Each of these sentences has already been translated into various languages. Thus, once standard sentences are chosen for each composed sentence, the entire document is simultaneously translated into each of the database languages. In this case, any complications related to the interpretation of grammar rules are obviated. If none of the standard sentences are appropriate for a particular sentence, program developers or the maintainers of the database could be contacted to expand the available choices. Clearly, as the system "learns", this option would be less frequently required.

One can most easily understand the process if one considers it as a generalization of a similar system, the Spell Checker in a word processor. There, when a questionable word is encountered, the Spell Checker presents its suggestions for possible correctly spelled words in descending order of merit. Similarly, this algorithm presents its permissible sentences in descending order of merit. If the word is correct, but was not recognized as such, the user can sometimes augment the Spell Checker dictionary.

The universality of the proposed organization is based on the realization of efficient facilities for approximate matching. Assuming that such facilities are available for human memory let us consider what kind of implications this may have on the design of the brain architecture. In the heart of computer science developments is the concept of a computational model: a set of rules for representation, accessing, and manipulation of information items. All reasonable computational models, according to the Church-Turing Thesis, are equivalent in their algorithmic capabilities. However, it appears to be impossible to trace the

information processing capabilities of the brain to a rudimentary Turing Machine. But is it easier to imagine how to implement a simpler artificial system, like a video game with a Turing Machine? For complex software systems, the key point is to find a convenient way of aggregating information, particularly in the way provided by the concept of object-oriented programming. For the case of the brain, to understand its information processing capabilities is to create an adequate computational model compatible with established or surmised hardware resources. Reduced to its essence, a computational model is an abstract construction that can drive memory contents from one state to another. Thus, it has two basic elementary characterizations: a way to access the memory and primitive procedures to transform the extracted information items. The Turing Machine, a computational model of pure theoretical significance, simply works on a sequential memory with singular symbols. A practical computational model, introduced by von Neumann, performs operations on a random access memory with word-organized transformations. Despite the tremendous advancements in computer technology the original von Neumann concept with auxiliary variations remains the prevailing principle in the design of most modern computers. Further evolution of computational models required a change in the way memory is accessed which led to the concept of content-addressable memory. The content-addressable memory is naturally applied to realization of approximate matching. With the approximate matching mechanism, certain crucial features of the organization of information processing in the brain can be readily elucidated.

As a subsequent step in the development of computational models we have suggested consideration of a computational model which incorporates a plurality of processors operating on a common content-addressable memory [5]. Besides some practical advantages, this computational model also raises a specific theoretical interest. It appears that this computational model "built around a global associative memory with a multiplicity of input points" exhibits a certain resemblance to the general organization of the brain. The characterization of the human brain as a computer device raises many confusing questions. Particularly, the main puzzle is the organization of the interaction between memory and processing elements. Experiments indicate that memory is distributed while processing elements are associated with definite localized areas of the brain. As such, how can a comparison of different types of information be organized with a slow transmission of neural pulses?

We employ the above computational model - a plurality of processors on a common content-addressable

memory, but with a "small" modification: we consider a very fast memory with relatively slow processors. Such a consideration goes against the grain of all technological trends. So, probably, it has not yet been considered from the point of view of computer engineering. The fast memory supplies a huge amount of information items, while the slow network of processing elements performs their ranking. As a result, with a "slow" interconnection structure this system can provide a coordinated activity among different processing elements. It looks as if the sophistication of the brain cannot be realized without an unimaginably complex computational mechanism. But this should not be the case. A proper computational model must rely on a single universal operational principle. A good algorithmic design is "like a sharp knife - it does exactly what it is supposed to do with a minimum amount of applied effort", while using a wrong design is "like trying to cut a steak with a screwdriver" [6]. For simple systems the effectiveness of algorithmic design may be just a question of aesthetic pleasing, but for complex systems, it is a question of feasibility.

Imagine all the variety of modern computer applications and notice how uncomplicated is the functionality of the underlying von Neumann's model:

(0) Set up an initial pointer to the list of instructions
(1) Fetch a machine instruction
(2) Take specified information items
(3) Perform indicated transformations
(4) Write down the results in the memory
(5) Determine the next instruction
(6) Go to (1)

Now, consider operational characteristics of the suggested computational model in view of its potential relationship with the information processing capabilities of the brain:

(0) Set up an initial state with starting information items
(1) Generate an access criterion and select information items
(2) Combine selected information items and sensory inputs
(3) Rank the obtained information items
(4) Modify a few top information items
(5) Write down certain information items in the memory
(6) Go to 1

This computational model appears to satisfy the moderate requirement of being algorithmically complete. In other words, there is no doubt that this model has the ability to perform all thinkable kinds of algorithms. The problem is how the qualities of the information processing conform to the properties of the human brain.

The holographic computational model arranges streams of memory information in cycles. This model does not need to have a hierarchical structure with a short-term and long-term memory. It contains a working memory of neuron structures and a permanent memory in the holographic medium. The permanent holographic memory operates like Write-Once Read-Many (WORM) memory; this happens by virtue of the scanning mechanism that does not direct writings back to previously referenced locations. Thus, human memory is permanent not because it is not erasable, but because its locations, having been used once, are not available for subsequent writings.

An information stream enters the brain and after a certain transformation, a particular part of it is written to the memory. This written part becomes a permanent component of the contents of the memory of an individual; it can join the information stream at the next access cycle. Appearance of short-term and long-term memory is determined by the changing priorities in access.

Initialization and termination of computations with von Neumann's model are straightforward: to start the computations it is necessary to indicate the beginning instruction as shown in line (0); an instruction to stop computations can be issued at line (5). In the holographic model the situation is more involved. The initialization of a purposeful information processing requires a kind of a bootstrapping to provide a particular set of starting information items. Thus, if for some reason the cycling of information streams is discontinued at a certain moment, the resumption of the information processing may require special activities. A termination condition for information processing in the holographic model is not foreseen. Actually, the termination occurs with a physical disintegration of the processing elements.

The human brain is definitely able to perform very diverse and sophisticated tasks. So, where is the software? Among the questions on the brain hardware structure this question is among the most perplexing. Sophisticated software is a decisive component of information processing, so where is it stored and how can it be acquired. Furthermore, brain apparently works in a multitasking mode, so how can cumbersome procedures such as the interrupt be managed.

The distinctive factor of the presented computational model is that its information processing is content-driven. Thus, this model integrates software and data

and the basic operational procedure in this arrangement is approximate matching in a plurality of information item collections.

References

1. Akaev A.A., Gurevich S.B., Zhumaliev K.M.
"Holographic memory",
Allerton Press, Inc., New York, 1998

2. Pribram K. "Languages of the Brain",
Brandon House, London, 1981

3. Berkovich S.Y. "On the Information
Processing Capabilities of the Brain: Shifting
the Paradigm", Nanobiology, Vol. 2, No 2,
pp. 99-107, 1993

4. Lapir G.M. "Use of associative access method
for information retrieval systems",
Proceedings of the 23rd Annual Pittsburgh
Conference on Modeling and Simulation, v.23,
part 2, pp. 951-958, Univ. of Pittsburgh, 1992

5. Berkovich S.Y. "A New Computational Model
for Massive Parallelism", Proceedings of
Frontiers '90, College Park, Maryland, IEEE
Computer Society, pp. 244-250, 1990

6. Cormen T.H., Leiserson C.E., Rivest R.L.
"Introduction to Algorithms", The MIT Press,
Cambridge, Massachusetts, 1990